Science texts (viruses) final stats imac

Code

Experiment 1: Viruses Text

Hide

```
library(afex)
```

```
Loading required package: lme4
Loading required package: Matrix
************
Welcome to afex. For support visit: http://afex.singmann.science/
- Functions for ANOVAs: aov_car(), aov_ez(), and aov_4()
- Methods for calculating p-values with mixed(): 'S', 'KR', 'LRT', and 'PB'
- 'afex_aov' and 'mixed' objects can be passed to emmeans() for follow-up tests
- NEWS: emmeans() for ANOVA models now uses model = 'multivariate' as default.
- Get and set global package options with: afex_options()
- Set orthogonal sum-to-zero contrasts globally: set_sum_contrasts()
- For example analyses see: browseVignettes("afex")
************

Attaching package: ‘afex’

The following object is masked from ‘package:lme4’:

    lmer
```

Hide

```
library(emmeans)
library(ltm)
```

```
Loading required package: MASS
Loading required package: msm
Loading required package: polycor
```

Hide

```
library(tidyverse)
```

```
Registered S3 methods overwritten by 'dbplyr':
  method         from
  print.tbl_lazy
  print.tbl_sql
── Attaching packages ─────────────────────────────────────────────────── tidyverse 1.3.2 ──✔ ggp
lot2 3.4.0      ✔ purrr   0.3.5
✔ tibble  3.1.8      ✔ dplyr   1.0.10
✔ tidyr   1.2.1      ✔ stringr 1.4.1
✔ readr   2.1.3      ✔ forcats 0.5.2 ── Conflicts ───────────────────────────────────────────────
─────── tidyverse_conflicts() ──
✖ tidyr::expand() masks Matrix::expand()
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
✖ tidyr::pack()   masks Matrix::pack()
✖ dplyr::select() masks MASS::select()
✖ tidyr::unpack() masks Matrix::unpack()
```

Hide

```
library(gt)
```

```
Registered S3 methods overwritten by 'htmltools':
  method               from
  print.html           tools:rstudio
  print.shiny.tag      tools:rstudio
  print.shiny.tag.list tools:rstudio
```

Hide

```
library(interactions)
library(cowplot)
library(broom.mixed)
library(gtsummary)
```

```

Attaching package: ‘gtsummary’

The following object is masked from ‘package:MASS’:

    select
```

Hide

```
home_dir = getwd()
df = read.csv(paste(home_dir,"MC_for_stats_in_r_n=170_11_8_21.csv",sep="/"),header=TRUE)
df$q_num = as.factor(df$q_num)
df$subjectGroup = recode(df$subjectGroup, "nsg:1"="RPm","nsg:2"="RPp", "nsg:3"= "NRP")
df$subjectGroup = factor(df$subjectGroup,levels=c("RPm","RPp","NRP"))

add = read.csv(paste(home_dir,"viruses_GMRT_familiarity_transformed.csv",sep="/"))
df = left_join(add,df,by="mturk_id")

df = df[,c("mturk_id","subjectGroup","GMRT_bc_c_s","familiarity_bc_c_s","q_num","q_type","correct")]
df = df %>% rename("Reading_Ability"= GMRT_bc_c_s,"Prior_Knowledge" = familiarity_bc_c_s)
```

Multiple Choice

Model Selection

Hide

```
# mm1.g = glmer(data=df,formula=(correct ~ subjectGroup * q_type * Reading_Ability * Prior_Knowledge  + (q_type|mturk_id) + (1|q_num)),family=binomial(link='logit'),control = glmerControl(optCtrl = list(maxfun = 1e6)))

# gm_all <- lme4::allFit(mm1.g) # almost all except Nelder_Mead (failed to converge) are singular

# mm2.g = glmer(data=df,formula=(correct ~ subjectGroup * q_type * Reading_Ability * Prior_Knowledge  + (q_type||mturk_id) + (1|q_num)),family=binomial(link='logit'), control = glmerControl(optCtrl = list(maxfun = 1e6)))

# gm_all2 <- lme4::allFit(mm2.g) # almost all except Nelder_Mead (failed to converge) are singular

# summary(mm2.g)$varcor
# # random slope estimate for q_type is very small

mm3.g = glmer(data=df,formula=(correct ~ subjectGroup * q_type * Reading_Ability * Prior_Knowledge + (1|mturk_id) + (1|q_num)),family=binomial(link='logit'), control = glmerControl(optCtrl = list(maxfun = 1e6),optimizer="bobyqa"))

# gm_all3 <- lme4::allFit(mm3.g) # 4 failed to converge

# check estimates of different optimizers, if they are practically equivalent, convergence warning is probably false positive
# Therefore use whichever converges fastest

# ss = summary(gm_all3)
# ss$ fixef               ## fixed effects
#   ss$ llik                ## log-likelihoods
#   ss$ sdcor               ## SDs and correlations
#   ss$ theta               ## Cholesky factors
#   ss$ which.OK            ## which fits worked

# Results are practically identical, therefore will proceed to use bobyqa and mm3.g as final model

# summary(mm3.g)
```

Multiple Choice Final Model

Hide

```
# no random slope for q_type
require(parallel)
```

```
Loading required package: parallel
```

Hide

```
cl <- makeCluster(rep("localhost", 6)) # make cluster
#
mm3 = afex::mixed(cl=cl,data=df,formula=(correct ~ subjectGroup * q_type * Reading_Ability * Prior
_Knowledge  + (1|mturk_id) + (1|q_num)), family=binomial(link='logit'),method="PB",args_test = lis
t(nsim = 1000, cl = cl),progress=TRUE,expand_re = TRUE,control = glmerControl(optCtrl = list(maxfu
n = 1e6),optimizer="bobyqa"))
```

```
Contrasts set to contr.sum for the following variables: subjectGroup, q_type, mturk_id, q_num
```

```
Fitting 16 (g)lmer() models.
Obtaining 15 p-values:
[...............]
```

Hide

```
stopCluster(cl)
```

Hide

```
# arm::binnedplot(fitted(mm3$full_model),
#            residuals(mm3$full_model, type = "response"),
#            nclass = NULL,
#            cex.pts = 0.8,
#            col.pts = 1,
#            col.int = "gray")
```

Hide

```
# check contrasts
contrasts(mm3$data$subjectGroup)
```

```
    [,1] [,2]
RPm    1    0
RPp    0    1
NRP   -1   -1
```

Hide

```
mm3
```

```
Mixed Model Anova Table (Type 3 tests, PB-method)

Model: correct ~ subjectGroup * q_type * Reading_Ability * Prior_Knowledge +
Model:     (1 | mturk_id) + (1 | q_num)
Data: df
                                               Effect df    Chisq p.value
1                                        subjectGroup  2     0.14    .939
2                                              q_type  1     2.12    .206
3                                     Reading_Ability  1 26.04 **    .001
4                                     Prior_Knowledge  1 10.94 **    .002
5                                 subjectGroup:q_type  2     0.73    .709
6                        subjectGroup:Reading_Ability  2   5.29 +    .097
7                              q_type:Reading_Ability  1   5.22 *    .024
8                        subjectGroup:Prior_Knowledge  2   7.05 *    .044
9                              q_type:Prior_Knowledge  1     1.92    .156
10                    Reading_Ability:Prior_Knowledge  1     1.86    .208
11                subjectGroup:q_type:Reading_Ability  2     0.17    .912
12                subjectGroup:q_type:Prior_Knowledge  2     0.84    .637
13       subjectGroup:Reading_Ability:Prior_Knowledge  2     1.39    .519
14             q_type:Reading_Ability:Prior_Knowledge  1     0.87    .320
15 subjectGroup:q_type:Reading_Ability:Prior_Knowledge  2     2.45    .296
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘+’ 0.1 ‘ ’ 1
```

Hide

```
summary(mm3$full_model)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod]
 Family: binomial  ( logit )
Formula: correct ~ subjectGroup * q_type * Reading_Ability * Prior_Knowledge +
    (1 | mturk_id) + (1 | q_num)
   Data: data
Control: glmerControl(optCtrl = list(maxfun = 1e+06), optimizer = "bobyqa")

     AIC      BIC   logLik deviance df.resid
  1587.2   1733.3   -767.6   1535.2     2014

Scaled residuals:
    Min      1Q  Median      3Q     Max
-6.2375  0.1477  0.2472  0.4030  2.5286

Random effects:
 Groups   Name        Variance Std.Dev.
 mturk_id (Intercept) 0.2835   0.5325
 q_num    (Intercept) 1.0981   1.0479
Number of obs: 2040, groups:  mturk_id, 170; q_num, 12

Fixed effects:
                                                      Estimate Std. Error z value Pr(>|z|)
(Intercept)                                            2.13689    0.32068   6.664 2.67e-11
subjectGroup1                                          0.01554    0.12346   0.126 0.899845
subjectGroup2                                         -0.04781    0.12709  -0.376 0.706765
q_type1                                               -0.47588    0.31497  -1.511 0.130822
Reading_Ability                                        0.46633    0.08912   5.233 1.67e-07
Prior_Knowledge                                        0.31265    0.09251   3.380 0.000726
subjectGroup1:q_type1                                  0.04123    0.10613   0.388 0.697655
subjectGroup2:q_type1                                 -0.09462    0.10914  -0.867 0.385986
subjectGroup1:Reading_Ability                          0.19527    0.12473   1.566 0.117439
subjectGroup2:Reading_Ability                         -0.28017    0.12393  -2.261 0.023781
q_type1:Reading_Ability                                0.17698    0.07589   2.332 0.019702
subjectGroup1:Prior_Knowledge                         -0.33715    0.12574  -2.681 0.007335
subjectGroup2:Prior_Knowledge                          0.21410    0.13543   1.581 0.113904
q_type1:Prior_Knowledge                                0.11220    0.07935   1.414 0.157335
Reading_Ability:Prior_Knowledge                       -0.12637    0.09130  -1.384 0.166352
subjectGroup1:q_type1:Reading_Ability                 -0.04316    0.10641  -0.406 0.685020
subjectGroup2:q_type1:Reading_Ability                  0.01067    0.10598   0.101 0.919786
subjectGroup1:q_type1:Prior_Knowledge                  0.09556    0.10791   0.886 0.375859
subjectGroup2:q_type1:Prior_Knowledge                 -0.02138    0.11568  -0.185 0.853350
subjectGroup1:Reading_Ability:Prior_Knowledge         -0.10838    0.12252  -0.885 0.376385
subjectGroup2:Reading_Ability:Prior_Knowledge         -0.03622    0.13393  -0.270 0.786820
q_type1:Reading_Ability:Prior_Knowledge               -0.07412    0.07789  -0.952 0.341300
subjectGroup1:q_type1:Reading_Ability:Prior_Knowledge -0.11403    0.10479  -1.088 0.276516
subjectGroup2:q_type1:Reading_Ability:Prior_Knowledge -0.05625    0.11377  -0.494 0.621020

(Intercept)                                           ***
subjectGroup1
subjectGroup2
q_type1
Reading_Ability                                       ***
Prior_Knowledge                                       ***
subjectGroup1:q_type1
```

```
subjectGroup2:q_type1
subjectGroup1:Reading_Ability
subjectGroup2:Reading_Ability                         *
q_type1:Reading_Ability                               *
subjectGroup1:Prior_Knowledge                         **
subjectGroup2:Prior_Knowledge
q_type1:Prior_Knowledge
Reading_Ability:Prior_Knowledge
subjectGroup1:q_type1:Reading_Ability
subjectGroup2:q_type1:Reading_Ability
subjectGroup1:q_type1:Prior_Knowledge
subjectGroup2:q_type1:Prior_Knowledge
subjectGroup1:Reading_Ability:Prior_Knowledge
subjectGroup2:Reading_Ability:Prior_Knowledge
q_type1:Reading_Ability:Prior_Knowledge
subjectGroup1:q_type1:Reading_Ability:Prior_Knowledge
subjectGroup2:q_type1:Reading_Ability:Prior_Knowledge
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Correlation matrix not shown by default, as p = 24 > 12.
Use print(x, correlation=TRUE)  or
    vcov(x)        if you need it
```

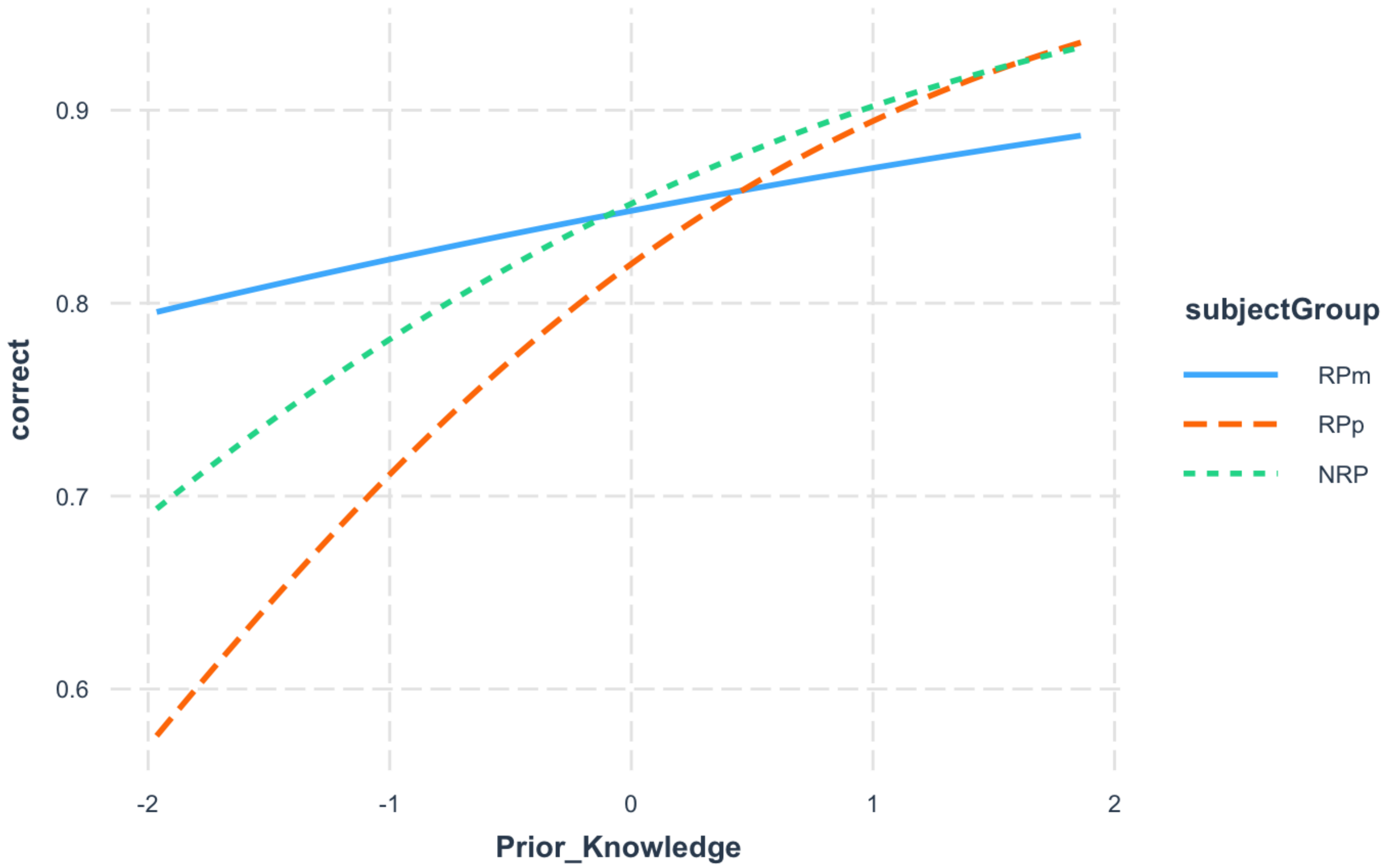
Interaction of subjectGroup and Prior Knowledge

Hide

```
probe_interaction(mm3$full_model, modx = subjectGroup, pred = Prior_Knowledge,plot.points = FALSE)
```

```
Warning: Johnson-Neyman intervals are not available for factor moderators.
```

```
SIMPLE SLOPES ANALYSIS

Slope of Prior_Knowledge when subjectGroup = NRP:

  Est.   S.E.   z val.      p
------ ------ -------- ------
  0.44   0.16     2.72   0.01

Slope of Prior_Knowledge when subjectGroup = RPp:

  Est.   S.E.   z val.      p
------ ------ -------- ------
  0.53   0.17     3.06   0.00

Slope of Prior_Knowledge when subjectGroup = RPm:

   Est.   S.E.   z val.      p
------- ------ -------- ------
  -0.02   0.15    -0.17   0.87
```



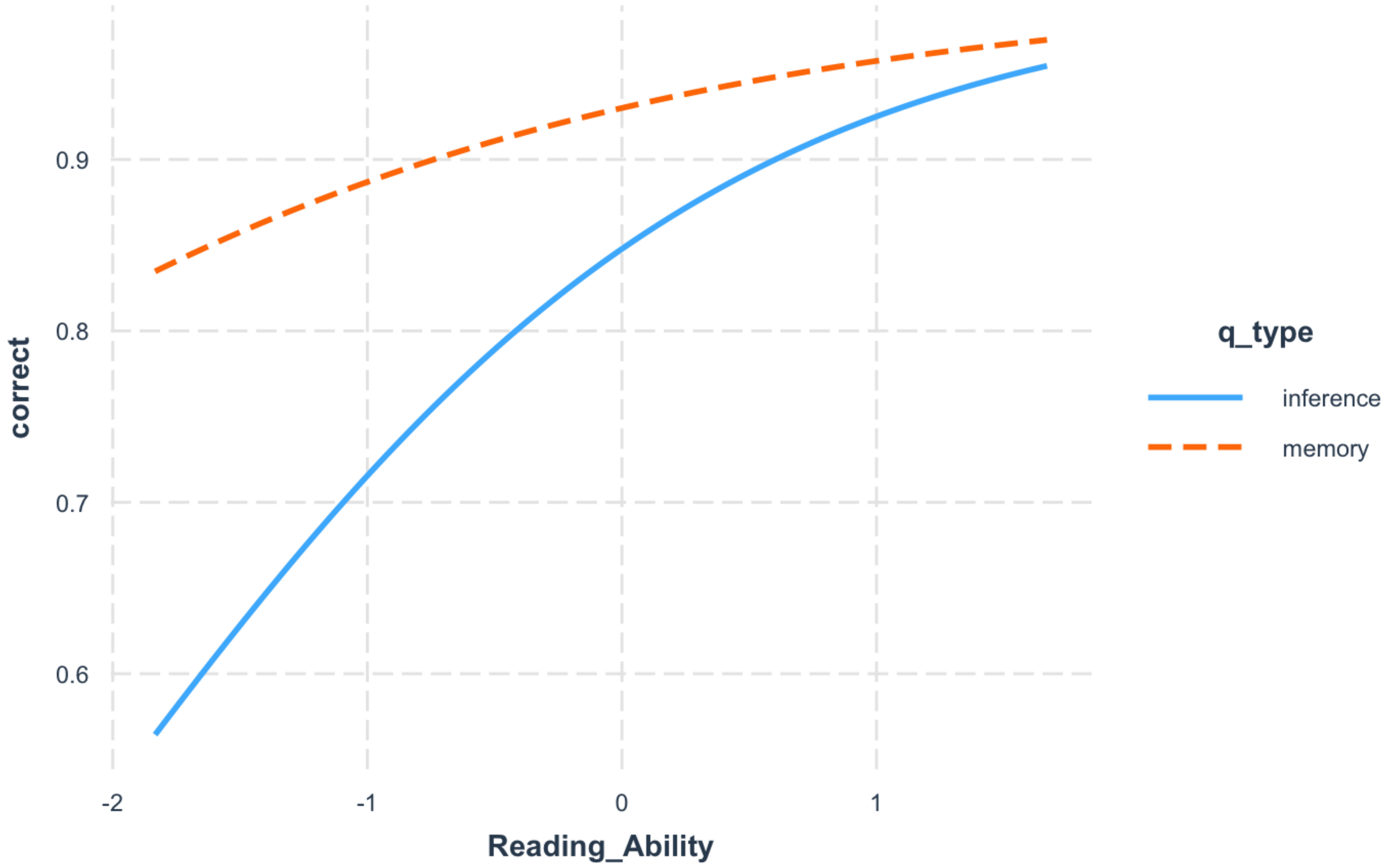
Interaction of question type and Reading Ability

Hide

```
probe_interaction(mm3$full_model, modx = q_type, pred = Reading_Ability,plot.points = FALSE)
```

```
Warning: Johnson-Neyman intervals are not available for factor moderators.
```

```
SIMPLE SLOPES ANALYSIS

Slope of Reading_Ability when q_type = memory:

  Est.   S.E.   z val.      p
------ ------ -------- ------
  0.29   0.13     2.30   0.02

Slope of Reading_Ability when q_type = inference:

  Est.   S.E.   z val.      p
------ ------ -------- ------
  0.64   0.11     5.98   0.00
```



Interaction of subjectGroup and Question Type

Hide

```
emm_options(glmer.df = "asymptotic")
emm_1 <- emmeans(mm3, "subjectGroup",by="q_type")
```

```
NOTE: Results may be misleading due to involvement in interactions
```

Hide

```
pairs(emm_1,adjust=NULL)
```

```
q_type = inference:
 contrast  estimate    SE  df z.ratio p.value
 RPm - RPp  0.19920 0.258 Inf   0.771  0.4408
 RPm - NRP -0.02889 0.257 Inf  -0.113  0.9104
 RPp - NRP -0.22809 0.260 Inf  -0.877  0.3806

q_type = memory:
 contrast  estimate    SE  df z.ratio p.value
 RPm - RPp -0.07250 0.313 Inf  -0.231  0.8171
 RPm - NRP -0.00458 0.301 Inf  -0.015  0.9879
 RPp - NRP  0.06792 0.313 Inf   0.217  0.8283

Results are given on the log odds ratio (not the response) scale.
```

Recall

Hide

```
home_dir = getwd()
df = read.csv(paste(home_dir,"binary_correct_n=170_10_18_22.csv",sep="/"),header=TRUE)
df$subjectGroup = recode(df$subjectGroup,"nsg:3"= "NRP", "nsg:1"="RPm","nsg:2"="RPp" )
df$subjectGroup = factor(df$subjectGroup, c("RPm","RPp","NRP"))

df$idea_units = as.factor(df$idea_units)

add = read.csv(paste(home_dir,"viruses_GMRT_familiarity_transformed.csv",sep="/"))
df = left_join(add,df,by="mturk_id")

df = df[,c("mturk_id","subjectGroup","GMRT_bc_c_s","familiarity_bc_c_s","idea_units","correct","RP_any","RP_imp","RP_per")]
df = df %>% rename("Reading_Ability"= GMRT_bc_c_s,"Prior_Knowledge" = familiarity_bc_c_s)
```

Main Idea Units

Hide

```
dfRPi = df[df$RP_imp == 1,]
```

Hide

```
# mm.RPi1.g = glmer(data=dfRPi,correct ~ subjectGroup * Reading_Ability * Prior_Knowledge  + (1|mturk_id) + (1|idea_units),family=binomial(link='logit'),control = glmerControl(optCtrl = list(maxfun = 1e6),optimizer = "bobyqa"))

# gm_all <- lme4::allFit(mm.RPi1.g) #

# ss <- summary(gm_all)
# ss$ fixef               ## fixed effects
# ss$ llik                ## log-likelihoods
# ss$ sdcor               ## SDs and correlations
# ss$ theta               ## Cholesky factors
# ss$ which.OK            ## which fits worked
# nearly identical, so will use boyqa for speed
```

Hide

```
require(parallel)
cl <- makeCluster(rep("localhost", 6)) # make cluster
mm.RPi1 = afex::mixed(cl=cl,data=dfRPi,formula=(correct ~ subjectGroup * Reading_Ability * Prior_Knowledge  + (1|mturk_id) + (1|idea_units)),family=binomial(link='logit'),method="PB",args_test = list(nsim = 1000, cl = cl),progress=TRUE,expand_re = TRUE,control = glmerControl(optCtrl = list(maxfun = 1e6),optimizer="bobyqa"))
```

```
Contrasts set to contr.sum for the following variables: subjectGroup, mturk_id, idea_units
```

```
Fitting 8 (g)lmer() models.
Obtaining 7 p-values:
[.......]
```

Hide

```
stopCluster(cl)
```

Hide

```
mm.RPi1
```

```
Mixed Model Anova Table (Type 3 tests, PB-method)

Model: correct ~ subjectGroup * Reading_Ability * Prior_Knowledge +
Model:     (1 | mturk_id) + (1 | idea_units)
Data: dfRPi
                                        Effect df    Chisq p.value
1                                 subjectGroup  2 23.49 **    .001
2                              Reading_Ability  1     0.10    .781
3                              Prior_Knowledge  1  7.49 **    .008
4                 subjectGroup:Reading_Ability  2     0.57    .777
5                 subjectGroup:Prior_Knowledge  2     1.83    .435
6              Reading_Ability:Prior_Knowledge  1   5.47 *    .025
7 subjectGroup:Reading_Ability:Prior_Knowledge  2     2.96    .244
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘+’ 0.1 ‘ ’ 1
```

Hide

```
# summary(mm.RPi1$full_model,correlation=FALSE)
```

Hide

```
emm_options(glmer.df = "asymptotic")
emm_1 <- emmeans(mm.RPi1, "subjectGroup")
```

```
NOTE: Results may be misleading due to involvement in interactions
```

Hide

```
pairs(emm_1,adjust=NULL)
```

```
 contrast  estimate    SE  df z.ratio p.value
 RPm - RPp     1.43 0.339 Inf   4.204  <.0001
 RPm - NRP     1.08 0.297 Inf   3.625  0.0003
 RPp - NRP    -0.35 0.362 Inf  -0.967  0.3336

Results are given on the log odds ratio (not the response) scale.
```

Peripheral Idea Units

Hide

```
dfRPp = df[df$RP_per == 1,]
```

Hide

```
mm.RPp1.g = glmer(data=dfRPp,correct ~ subjectGroup * Reading_Ability * Prior_Knowledge  + (1|mturk_id) + (1|idea_units),family=binomial(link='logit'),control = glmerControl(optCtrl = list(maxfun = 1e6),optimizer = "bobyqa"))

# gm_all <- lme4::allFit(mm.RPp1.g) # 4 failed to converge, compare results

# ss <- summary(gm_all)
# ss$ fixef               ## fixed effects
# ss$ llik                ## log-likelihoods
# ss$ sdcor               ## SDs and correlations
# ss$ theta               ## Cholesky factors
# ss$ which.OK            ## which fits worked
# nearly identical, so will use bobyqa for speed
```

Hide

```
require(parallel)
cl <- makeCluster(rep("localhost", 6)) # make cluster
mm.RPp1 = afex::mixed(cl=cl,data=dfRPp,formula=(correct ~ subjectGroup * Reading_Ability * Prior_Knowledge  + (1|mturk_id) + (1|idea_units)),family=binomial(link='logit'),method="PB",args_test = list(nsim = 1000, cl = cl),progress=TRUE,expand_re = TRUE,control = glmerControl(optCtrl = list(maxfun = 1e6),optimizer="bobyqa"))
```

```
Contrasts set to contr.sum for the following variables: subjectGroup, mturk_id, idea_units
```

```
Fitting 8 (g)lmer() models.
Obtaining 7 p-values:
[.......]
```

Hide

```
stopCluster(cl)
```

Hide

```
mm.RPp1
```

```
Mixed Model Anova Table (Type 3 tests, PB-method)

Model: correct ~ subjectGroup * Reading_Ability * Prior_Knowledge +
Model:     (1 | mturk_id) + (1 | idea_units)
Data: dfRPp
                                        Effect df    Chisq p.value
1                                 subjectGroup  2 84.59 **    .001
2                              Reading_Ability  1     2.92    .105
3                              Prior_Knowledge  1  7.43 **    .007
4                 subjectGroup:Reading_Ability  2     0.16    .915
5                 subjectGroup:Prior_Knowledge  2     0.23    .901
6              Reading_Ability:Prior_Knowledge  1     1.45    .258
7 subjectGroup:Reading_Ability:Prior_Knowledge  2     1.41    .514
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘+’ 0.1 ‘ ’ 1
```

Hide

```
# summary(mm.RPp1$full_model,correlation=FALSE)
```

Hide

```
emm_options(glmer.df = "asymptotic")
emm_1 <- emmeans(mm.RPp1, "subjectGroup")
```

```
NOTE: Results may be misleading due to involvement in interactions
```

Hide

```
pairs(emm_1,adjust=NULL)
```

```
 contrast  estimate    SE  df z.ratio p.value
 RPm - RPp   -3.046 0.373 Inf  -8.169  <.0001
 RPm - NRP   -0.597 0.397 Inf  -1.505  0.1323
 RPp - NRP    2.449 0.342 Inf   7.170  <.0001

Results are given on the log odds ratio (not the response) scale.
```

Non-practiced Idea Units

Hide

```
dfNoRP = df[df$RP_any == 0,]
mm.noRP1.g = glmer(data=dfNoRP,correct ~ subjectGroup * Reading_Ability * Prior_Knowledge + (1|mturk_id) + (1|idea_units),family=binomial(link='logit'),control = glmerControl(optCtrl = list(maxfun = 1e6),optimizer = "bobyqa"))

# gm_all <- lme4::allFit(mm.noRP1.g) # 2 failed to converge, compare results
#
# ss <- summary(gm_all)
#   ss$ fixef              ## fixed effects
#   ss$ llik               ## log-likelihoods
#   ss$ sdcor              ## SDs and correlations
#   ss$ theta              ## Cholesky factors
#   ss$ which.OK           ## which fits worked
# nearly identical, so will use boyqa for speed
```

Hide

```
# no random slope for q_type
require(parallel)
cl <- makeCluster(rep("localhost", 6)) # make cluster
#
mm.noRP1 = afex::mixed(cl=cl,data=dfNoRP,formula=(correct ~ subjectGroup * Reading_Ability * Prior_Knowledge + (1|mturk_id) + (1|idea_units)), family=binomial(link='logit'),method="PB",args_test = list(nsim = 1000, cl = cl),progress=TRUE,expand_re = TRUE,control = glmerControl(optCtrl = list(maxfun = 1e6),optimizer="bobyqa"))
```

```
Contrasts set to contr.sum for the following variables: subjectGroup, mturk_id, idea_units
```

```
Fitting 8 (g)lmer() models.
Obtaining 7 p-values:
[
```

Hide

```
mm.noRP1
```

```
Mixed Model Anova Table (Type 3 tests, PB-method)

Model: correct ~ subjectGroup * Reading_Ability * Prior_Knowledge +
Model:     (1 | mturk_id) + (1 | idea_units)
Data: dfNoRP
                                      Effect df    Chisq p.value
1                               subjectGroup  2   6.20 +    .082
2                            Reading_Ability  1 11.38 **    .003
3                            Prior_Knowledge  1     2.81    .116
4               subjectGroup:Reading_Ability  2     2.90    .249
5               subjectGroup:Prior_Knowledge  2     4.44    .135
6            Reading_Ability:Prior_Knowledge  1     1.41    .259
7 subjectGroup:Reading_Ability:Prior_Knowledge  2     3.75    .177
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘+’ 0.1 ‘ ’ 1
```

Hide

```
summary(mm.noRP1$full_model)
```

```
Generalized linear mixed model fit by maximum likelihood (Laplace Approximation) [glmerMod]
 Family: binomial  ( logit )
Formula: correct ~ subjectGroup * Reading_Ability * Prior_Knowledge +  
    (1 | mturk_id) + (1 | idea_units)
   Data: data
Control: glmerControl(optCtrl = list(maxfun = 1e+06), optimizer = "bobyqa")

     AIC      BIC   logLik deviance df.resid 
  4261.1   4353.1  -2116.6   4233.1     5256 

Scaled residuals: 
    Min      1Q  Median      3Q     Max 
-2.1601 -0.4418 -0.2907 -0.1650  6.5558 

Random effects:
 Groups     Name        Variance Std.Dev.
 mturk_id   (Intercept) 0.6298   0.7936  
 idea_units (Intercept) 0.8978   0.9475  
Number of obs: 5270, groups:  mturk_id, 170; idea_units, 31

Fixed effects:
                                              Estimate Std. Error z value Pr(>|z|)    
(Intercept)                                   -2.08195    0.19098 -10.901  < 2e-16 ***
subjectGroup1                                  0.12778    0.11249   1.136 0.255994    
subjectGroup2                                 -0.30040    0.12008  -2.502 0.012361 *  
Reading_Ability                                0.28197    0.08205   3.437 0.000589 ***
Prior_Knowledge                                0.14497    0.08558   1.694 0.090297 .  
subjectGroup1:Reading_Ability                 -0.14389    0.11489  -1.252 0.210420    
subjectGroup2:Reading_Ability                  0.19174    0.11674   1.642 0.100503    
subjectGroup1:Prior_Knowledge                 -0.07614    0.11423  -0.666 0.505100    
subjectGroup2:Prior_Knowledge                 -0.17712    0.12896  -1.373 0.169612    
Reading_Ability:Prior_Knowledge                0.10210    0.08509   1.200 0.230143    
subjectGroup1:Reading_Ability:Prior_Knowledge  0.02634    0.11287   0.233 0.815456    
subjectGroup2:Reading_Ability:Prior_Knowledge  0.19770    0.12798   1.545 0.122407    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Correlation of Fixed Effects:
            (Intr) sbjcG1 sbjcG2 Rdng_A Prr_Kn sbG1:R_A sbG2:R_A sG1:P_ sG2:P_ R_A:P_
subjectGrp1 -0.033                                                                   
subjectGrp2  0.053 -0.526                                                            
Redng_Ablty -0.018  0.142 -0.180                                                     
Prir_Knwldg -0.027 -0.029  0.150 -0.349                                              
sbjctG1:R_A  0.064  0.073  0.047 -0.029 -0.016                                       
sbjctG2:R_A -0.083  0.048 -0.151  0.021 -0.086 -0.497                                
sbjctG1:P_K -0.012 -0.080 -0.066 -0.015 -0.162 -0.386    0.261                       
sbjctG2:P_K  0.066 -0.062  0.054 -0.084  0.180  0.235   -0.386   -0.525              
Rdng_Ab:P_K -0.145  0.010 -0.105 -0.130  0.095 -0.049    0.110    0.164 -0.067       
sG1:R_A:P_K  0.004 -0.358  0.243 -0.051  0.166 -0.177    0.023    0.231 -0.107 -0.181
sG2:R_A:P_K -0.047  0.229 -0.373  0.105 -0.066  0.019   -0.049   -0.106  0.038  0.174
            sG1:R_A:
subjectGrp1         
subjectGrp2         
Redng_Ablty         
```

```
Prir_Knwldg
sbjctG1:R_A
sbjctG2:R_A
sbjctG1:P_K
sbjctG2:P_K
Rdng_Ab:P_K
sG1:R_A:P_K
sG2:R_A:P_K -0.512
```

Hide

```
emm_options(glmer.df = "asymptotic")
emm_1 <- emmeans(mm.noRP1, "subjectGroup")
```

```
NOTE: Results may be misleading due to involvement in interactions
```

Hide

```
pairs(emm_1,adjust=NULL)
```

```
 contrast  estimate    SE  df z.ratio p.value
 RPm - RPp   0.4282 0.203 Inf   2.107  0.0351
 RPm - NRP  -0.0448 0.191 Inf  -0.234  0.8147
 RPp - NRP  -0.4730 0.205 Inf  -2.311  0.0208

Results are given on the log odds ratio (not the response) scale.
```

Hide

```
# summary(mm.noRP1$full_model,correlation=FALSE)
```

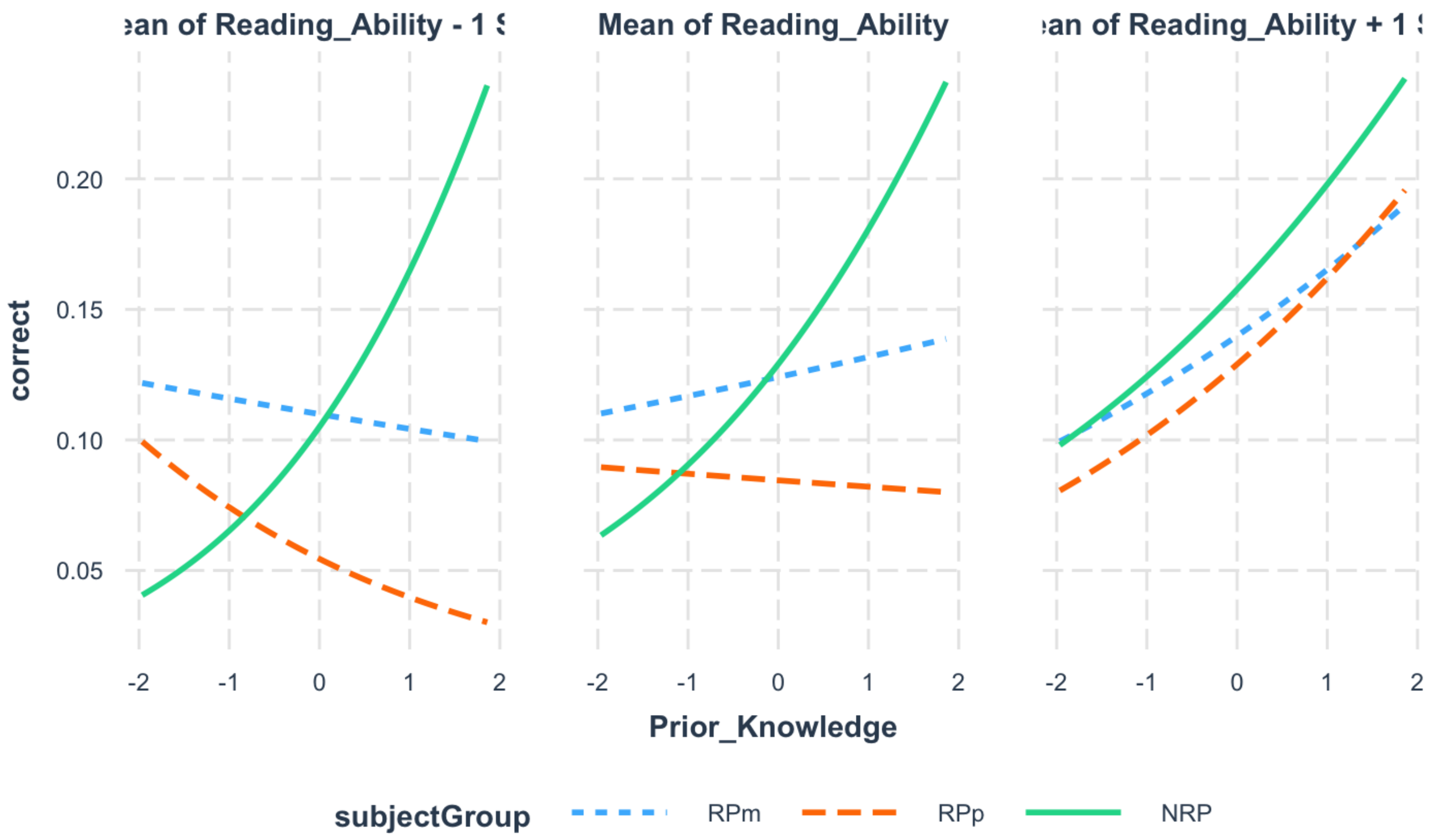
Hide

```
probe_interaction(mm.noRP1$full_model, pred = Prior_Knowledge, modx = subjectGroup,mod2 = Reading_
Ability , plot.points = FALSE)
```

```
Warning: Johnson-Neyman intervals are not available for factor moderators.
```

```
[redacted] While Reading_Ability (2nd moderator) = -9.971491e-01 (- 1 SD) [redacted]

SIMPLE SLOPES ANALYSIS

Slope of Prior_Knowledge when subjectGroup = RPm:

   Est.   S.E.   z val.      p
------- ------ -------- ------
  -0.06   0.14    -0.42   0.67

Slope of Prior_Knowledge when subjectGroup = RPp:

   Est.   S.E.   z val.      p
------- ------ -------- ------
  -0.33   0.24    -1.41   0.16

Slope of Prior_Knowledge when subjectGroup = NRP:

  Est.   S.E.   z val.      p
------ ------ -------- ------
  0.52   0.21     2.51   0.01

[redacted] While Reading_Ability (2nd moderator) = -1.929039e-16 (Mean) [redacted]

SIMPLE SLOPES ANALYSIS

Slope of Prior_Knowledge when subjectGroup = RPm:

  Est.   S.E.   z val.      p
------ ------ -------- ------
  0.07   0.13     0.52   0.60

Slope of Prior_Knowledge when subjectGroup = RPp:

   Est.   S.E.   z val.      p
------- ------ -------- ------
  -0.03   0.17    -0.19   0.85

Slope of Prior_Knowledge when subjectGroup = NRP:

  Est.   S.E.   z val.      p
------ ------ -------- ------
  0.40   0.14     2.77   0.01

[redacted] While Reading_Ability (2nd moderator) =  9.971491e-01 (+ 1 SD) [redacted]

SIMPLE SLOPES ANALYSIS

Slope of Prior_Knowledge when subjectGroup = RPm:

  Est.   S.E.   z val.      p
------ ------ -------- ------
  0.20   0.22     0.91   0.37
```

```
Slope of Prior_Knowledge when subjectGroup = RPp:

  Est.   S.E.   z val.      p
------ ------ -------- ------
  0.27   0.23     1.14   0.25

Slope of Prior_Knowledge when subjectGroup = NRP:

  Est.   S.E.   z val.      p
------ ------ -------- ------
  0.28   0.20     1.37   0.17
```


ean of Reading_Ability - 1 S
Mean of Reading_Ability
ean of Reading_Ability + 1 S
correct
Prior_Knowledge
subjectGroup
RPm
RPp
NRP